IJCCS, Vol.9, No.2, July 2015, pp. 157~166
ISSN: 1978-1520

Metode RCE-Kmeans untuk Clustering Data

Izmy Alwiah Musdar*[1], Azhari SN[2]
[1]Program Studi S2 Ilmu Komputer, FMIPA UGM, Yogyakarta
[2]Jurusan Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta
e-mail: ***[1]izmyalwiah@gmail.com**, [2]arisn.softcomp@gmail.com

Abstrak

Telah banyak metode yang dikembangkan untuk memecahkan berbagai masalah clustering. Salah satunya menggunakan metode-metode dari bidang kecerdasan kelompok seperti Particle Swarm Optimization (PSO). Metode Rapid Centroid Estimation (RCE) merupakan salah satu metode clustering yang berbasis PSO. RCE, seperti varian PSO clustering lainnya, memiliki kelebihan yaitu hasil clustering tidak tergantung pada inisialisasi pusat cluster awal. RCE juga memiliki waktu komputasi yang jauh lebih cepat dibandingkan dengan metode sebelumnya yaitu Particle Swarm Clustering (PSC) dan modified Particle Swarm Clustering (mPSC), tetapi metode RCE memiliki standar deviasi kualitas skema clustering yang lebih tinggi dibandingkan PSC dan mPSC dimana ini berpengaruh terhadap variansi hasil clustering. Hal ini terjadi karena equilibrium state, yaitu kondisi dimana posisi partikel tidak mengalami perubahan lagi, kurang tepat pada saat kriteria berhenti tercapai. Penelitian ini mengusulkan metode RCE-Kmeans yaitu metode yang mengaplikasikan K-means setelah equilibrium state metode RCE tercapai untuk memperbarui posisi partikel yang dihasilkan dari metode RCE. Hasil penelitian menunjukkan bahwa dari sepuluh dataset, metode RCE-Kmeans memiliki nilai kualitas skema clustering yang lebih baik pada 7 dataset dibandingkan K-means dan lebih baik pada 8 dataset dibandingkan dengan metode RCE. Penggunaan K-means pada metode RCE juga mampu menurunkan nilai standar deviasi dari metode RCE.

Kata kunci*—Clustering Data, Particle Swarm, K-means, Rapid Centroid Estimation.*

Abstract

There have been many methods developed to solve the clustering problem. One of them is method in swarm intelligence field such as Particle Swarm Optimization (PSO). Rapid Centroid Estimation (RCE) is a method of clustering based Particle Swarm Optimization. RCE, like other variants of PSO clustering, does not depend on initial cluster centers. Moreover, RCE has faster computational time than the previous method like PSC and mPSC. However, RCE has higher standar deviation value than PSC and mPSC in which has impact in the variance of clustering result. It is happaned because of improper equilibrium state, a condition in which the position of the particle does not change anymore, when the stopping criteria is reached. This study proposes RCE-Kmeans which is a method applying K-means after the equilibrium state of RCE reached to update the particle's position which is generated from the RCE method. The results showed that RCE-Kmeans has better quality of the clustering scheme in 7 of 10 datasets compared to K-means and better in 8 of 10 dataset then RCE method. The use of K-means clustering on the RCE method is also able to reduce the standard deviation from RCE method.

Keywords*—Data Clustering, Particle Swarm, K-means, Rapid Centroid Estimation.*

Received September 24th,2014; Revised January 15th, 2015; Accepted July 10th, 2015

1. PENDAHULUAN

Clustering merupakan sebuah teknik pemrosesan data yang digunakan untuk menemukan pola-pola tersembunyi pada kumpulan data [1]. *Clustering* telah banyak diaplikasikan dalam berbagai bidang antara lain penambangan data, pengenalan pola, pengambilan keputusan, *machine learning*, dan segmentasi citra [2]. Proses penemuan pola data dilakukan dengan mengelompokkan data ke dalam klaster-klaster sehingga data-data yang memiliki kemiripan berada pada klaster yang sama dan data-data yang tidak memiliki kemiripan terletak pada klaster yang berbeda [3]. Salah satu cara untuk mengetahui tingkat kemiripan data adalah melalui perhitungan jarak antara data. Semakin kecil jarak antar data maka semakin tinggi tingkat kemiripan data tersebut dan sebaliknya semakin besar jarak antar data maka semakin rendah tingkat kemiripannya.

K-means dan variannya merupakan jenis algoritma *clustering partition-based* yang telah banyak digunakan dalam *clustering* data [4]. K-means mengelompokkan kumpulan data ke dalam k klaster berdasarkan jarak terdekat antara data dengan pusat klaster [1]. Kelebihan dari algoritma K-means terletak pada kecepatan untuk mencapai konvergen serta kemudahan dalam pengimplementasian [5]. Pada sisi lain, algoritma K-means memiliki beberapa kelemahan seperti: (i) kecenderungan mengalami konvergensi prematur pada *quantization error* yang besar [6]; (ii) hasil clustering yang sangat bergantung pada penentuan pusat klaster awal [7,8]; serta (iii) mengalami masalah *dead-unit* [1].

Particle Swarm Optimization (PSO) merupakan jenis algoritma evolusi yang terinspirasi dari kawanan burung dan kawanan ikan [9]. Walaupun pada awalnya metode PSO dibuat untuk penyelesaian masalah optimasi, beberapa tahun terakhir metode PSO telah banyak diaplikasikan untuk memecahkan berbagai masalah yang berkaitan dengan *clustering* [10]. Hal ini disebabkan karena PSO mampu memberikan hasil *clustering* yang lebih stabil karena tidak adanya ketergantungan pada inisialisai pusat klaster awal [3]. Tetapi PSO juga memiliki kelemahan pada kecepatan konvergensinya, yaitu cenderung lambat saat mendekati solusi optimum [11].

Penerapan PSO untuk *clustering* data pertama kali dilakukan oleh Van der Merwe dan Engelbrecht [6]. Proses *clustering* mula-mula dilakukan dengan K-means yang kemudian dilanjutkan oleh PSO. Hasil *clustering* dari metode K-means digunakan sebagai salah satu partikel awal pada metode PSO. Penggunaan hasil *clustering* K-means sebagai salah satu partikel awal PSO ternyata mampu meningkatkan performansi dari PSO *clustering*.

Variasi lain dari Particle Swarm Optimization untuk melakukan *clustering* data adalah *Particle Swarm Clustering* (PSC) [12]. Berbeda dari metode PSO yang merepresentasikan setiap partikel sebagai satu himpunan pusat klaster, pada PSC setiap partikelnya cukup merepresentasikan satu pusat klaster saja. Sehingga solusi akhir klaster-klaster dari pendekatan PSC diperoleh dengan menggabungkan keseluruhan partikel yang ada. Hasil penelitian ini menunjukkan bahwa performansi metode PSC lebih unggul dibandingkan dengan K-means karena PSC dapat terhindar dari stagnasi.

Modified PSC (mPSC) merupakan metode yang diusulkan oleh [13] yang bertujuan untuk mempercepat kecepatan komputasi metode PSC. Metode mPSC mengusulkan ide mengganti velocity (V) dengan Δx, untuk mengeliminasi kebutuhan akan bobot inersia (ω). Dengan demikian, metode mPSC terbukti memiliki waktu komputasi yang sedikit lebih cepat dibandingkan dengan PSC.

Metode Rapid Centroid Estimation (RCE) [14] merupakan metode yang diusulkan untuk meningkatkan kinerja PSC dan mPSC. RCE memodifikasi metode mPSC pada bagian frekuensi pembaruan posisi partikel, frekuensi pembaruan matriks jarak (partikel dan titik data) dan partikel terbaik, dan menambahkan *global minimum computation* untuk penyimpanan kombinasi posisi partikel terbaik. Metode RCE melakukan *clustering* dengan waktu komputasi yang jauh lebih cepat dibandingkan dengan kedua metode sebelumnya tanpa mempengaruhi

kualitas skema clustering. Namun demikian, dibandingkan dengan PSC dan mPSC, standar deviasi kualitas skema clustering yang dihasilkan dari RCE lebih tinggi walaupun rata-rata hasil *clustering* RCE lebih unggul. Menurut [14], hal tersebut terjadi karena penentuan kriteria berhenti yang didasarkan pada tercapainya *equilibrium state* adalah kriteria berhenti yang kurang tepat.

Pada penelitian ini digunakan metode clustering RCE-Kmeans. K-means akan dijadikan metode yang mendefinisikan kembali *equilibrium state* dari metode clustering RCE. Oleh karena itu, posisi partikel akhir yang merepresentasikan pusat klaster diperoleh setelah penerapan metode K-means. K-means dipilih karena memiliki kemampuan *local search* [15], yaitu kemampuan menemukan solusi optimum yang berada di sekitar nilai solusi awal yang didefinisikan. Kemampuan *local search* yang dimiliki K-means dibutuhkan karena kemampuan *local search* menjamin penemuan solusi optimum (posisi partikel optimum) di sekitar nilai solusi awal (posisi partikel berdasarkan *equilibrium state* RCE) bukan mencari ruang solusi baru yang memungkinkan diperoleh hasil clustering yang tidak lebih baik dari yang dihasilkan oleh RCE. K-means juga memiliki kelebihan pada kecepatan konvergensinya sehingga diharapkan penggunaan metode K-means untuk menemukan posisi partikel optimum dari metode RCE tidak menambah waktu komputasi secara signifikan. Dengan demikian, penggunaan metode K-means dalam penentuan posisi partikel akhir metode RCE mampu menghasilkan posisi partikel yang dapat meningkatkan kualitas skema hasil clustering dan mampu memperkecil pengaruh *equlibrium state* terhadap kualitas skema clustering yang dilihat dari nilai standar deviasi kualitas skema clustering.

2. METODE PENELITIAN

RCE-Kmeans merupakan metode clustering yang diusulkan untuk menyelesaikan masalah metode RCE. Metode K-means digunakan setelah kriteria berhenti metode RCE terpenuhi. Posisi partikel akhir yang diperoleh dari RCE kemudian digerakkan menggunakan metode K-means. Alur proses dari metode RCE-Kmeans ditunjukkan pada Gambar 1 dengan penjelasan sebagai berikut :

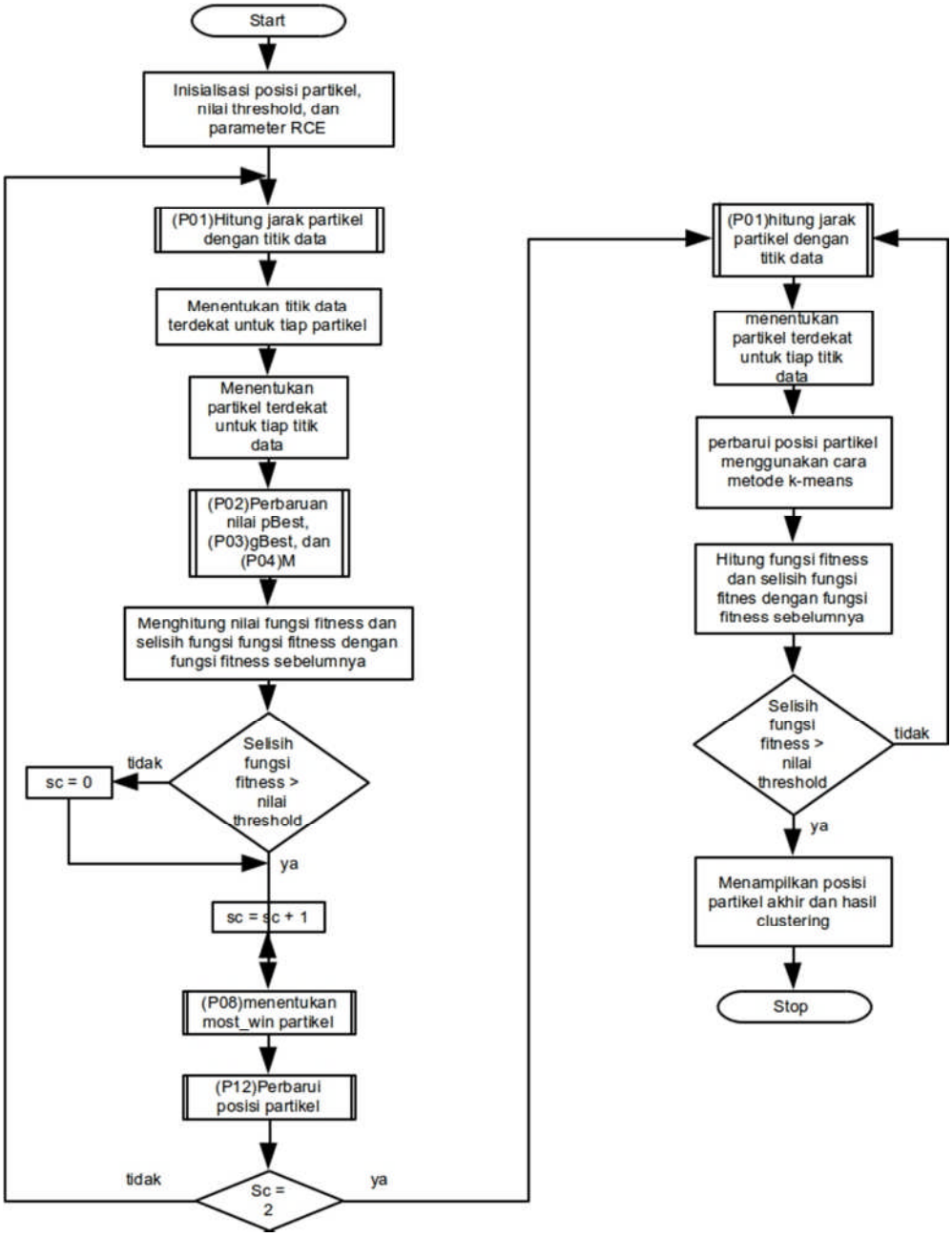
1. Inisialisasi beberapa parameter yang dibutuhkan. Parameter tersebut adalah bobot inersia mula-mula, nilai *threshold*, bobot yang mempengaruhi nilai faktor kognitif, sosial, dan *self-organizing* (masing-masing φ_1, φ_2, φ_3), posisi partikel mula-mula, dan jumlah partikel. Untuk jumlah partikel disesuaikan dengan jumlah kelas dari dataset yang akan melalui proses clustering. Nilai dari masing-masing parameter yang digunakan disesuaikan dengan nilai parameter pada penelitian [14]. Nilai dari masing-masing parameter adalah jumlah Percobaan = 50, $\omega_{\text{mula-mula}}$ = 0.9, *Decay Rate* = 0.95, threshold = 0.00001. Sedangkan untuk nilai φ_1, φ_2, φ_3 diperoleh dengan mencoba berbagai kombinasi nilai φ_1, φ_2, φ_3 dan dipilih kombinasi yang menghasilkan kualitas skema clustering terbaik. Kombinasi nilai φ_1, φ_2, φ_3 yang digunakan ditunjukkan pada Tabel 1.
2. Menghitung jarak tiap partikel dengan tiap titik data. Penghitungan jarak menggunakan persamaan (1).

$$, = \sqrt{\sum (-)} \quad (1)$$

   dimana $\mathbf{x_p}$ adalah titik data ke-p, $\mathbf{z_j}$ melambangkan titik pusat klaster ke-j, dan d merupakan jumlah atribut dari tiap pusat klaster.
3. Menentukan titik data terdekat untuk masing-masing partikel dan partikel terdekat untuk masing-masing titik data. Langkah untuk memperoleh titik data yang paling dekat dengan suatu partikel adalah dengan membandingkan jarak antara masing-masing data terhadap partikel tersebut. Titik data yang memiliki jarak terkecil akan ditandai sebagai titik data

terdekat. Sedangkan untuk menentukan partikel terdekat untuk masing-masing titik data dilakukan sebaliknya, yaitu membandingkan jarak masing-masing partikel terhadap suatu titik data. Partikel yang memiliki jarak terdekat dengan titik data tersebut akan ditandai sebagai partikel terdekat.

4. Menentukan nilai pbest, gbest, dan m. Nilai pbest merepresentasikan posisi terbaik yang pernah dilalui oleh masing-masing partikel, gbest merepresentasikan partikel dengan posisi terdekat dari setiap titik data dan m merepresentasikan kombinasi posisi partikel terbaik. Kombinasi posisi partikel yang memiliki nilai fungsi fitness terkecil yang dinyatakan sebagai kombinasi posisi partikel terbaik. Nilai pbest, gbest dan m ini berpengaruh pada penentuan besarnya perpindahan partikel.



Gambar 1 Alur Proses Metode RCE-Kmeans

Tabel 1 Kombinasi paramter $\varphi_1, \varphi_2, \varphi_3$ untuk setiap dataset

Dataset	φ_1	φ_2	φ_3
Iris	0.5	0.7	0.8
Wine	0.9	0.4	0.9
Glass	0.9	0.7	0.7
Dermatology	0.1	0.5	0.9
WDBC	0.9	0.3	0.6
CMC	0.2	0.8	0.8
Yeast	0.7	0.3	0.8
Texture	0.2	0.6	0.9
Optical Digits	0.6	0.3	0.3
Thyroid	0.8	0.3	0.2

5. Menghitung fungsi fitness. Fungsi fitness yang digunakan adalah *sum of Euclidean distance*. Fungsi fitness digunakan sebagai penentu nilai stopping criteria. Jika nilai selisih fungsi fitness dari iterasi sebelumnya dan nilai fitness sekarang lebih kecil dari nilai threshold yang sudah ditetapkan maka nilai stopping criteria di-*increment-kan*.
6. Penentuan *most-won particle. Most-won particle* merupakan partikel dengan jarak paling dekat terhadap suatu titik data dibandingkan dengan jarak partikel-partikel lainnya terhadap suatu titik data. *Most-won particle* berpengaruh pada besarnya perpindahan partikel yang tidak pernah mengalami *win.*
7. Perbaruan posisi partikel pada metode RCE dipengaruhi oleh 3 faktor yaitu faktor kognitif, sosial dan Self-organizing. Penghitungan besarnya perpindahan partikel didahului dengan menghitung nilai ketiga faktor tersebut. Penghitungan faktor kognitif, sosial, dan self-organizing masing-masing dilakukan menggunakan persamaan (2), persamaan (3), dan persamaan (4).

$$= ( ) - ( ) \quad (2)$$

dimana X_i adalah faktor kognitif partikel ke-i, $x_i(t)$ melambangkan posisi partikel ke-i dan $p_i(t)$ melambangkan posisi terbaik dari partikel i terhadap titik data j.

$$( ) = \frac{\Sigma\forall \quad ( ) \quad \otimes(\quad ( ) \quad ( ))}{} \quad (3)$$

dimana Y_i adalah faktor sosial partikel ke-i, $\varphi_{i,j}$ merupakan nilai random yang melambangkan tingkat subjektifitas terhadap pola input, nilainya berkisar antara 0 sampai 1, $g_j(t)$ merepresentasikan posisi terdekat input data j dengan suatu partikel, dan N_j adalah jumlah data yang berjarak terdekat dengan partikel ke-i.

$$( ) = \frac{\Sigma\forall \quad ( ) \quad \otimes(\quad ( ) \quad ( ))}{} \quad (4)$$

dimana Z_i adalah faktor *self-organizing* yang mempengaruhi partikel ke-i, y_j melambangkan titik data ke-j, dan $\varphi_{i,j}$ yang merupakan variabel yang bernilai antara 0 sampai dengan 1 .

Penghitungan small-perturbation yang menyatakan besarnya perpindahan partikel. Penghitungan nilai small-perturbation dilakukan berdasarkan persamaan (5). Penentuan posisi partikel dilakukan menggunakan persamaan (6).

$$\Delta \ (\ + 1) = \quad ( ) \Delta \quad + \quad \otimes \quad ( ) + \quad ( ) + \quad ( ) \quad (5)$$

dimana $w(t)$ adalah bobot inersia. RCE menginisialisasi bobot inersia dengan 0.9 kemudian

akan berangsung-angsur berkurang setiap iterasinya, Δx_i melambangkan *small perturbation* atau besarnya perpidahan yang akan dilakukan oleh partikel ke-i, dan x_i melambangkan posisi partikel ke-i.

$$(+ 1) = ( ) + \Delta (+ 1) \quad (6)$$

Jika terdapat partikel yang tidak memiliki titik data yang terletak dengannya maka perhitungan besarnya perpindahan dilakukan menggunakan persamaan 7.

$$\Delta (+ 1) = \otimes (( ) - ( )) \quad (7)$$

dimana x_{most_win} merepresentasikan posisi partikel yang memiliki jarak terdekat dengan suatu titik data, dan φ_5 memiliki nilai antara 0 dan 1.

8. Setelah variabel stopping criteria bernilai 2 maka proses clustering dilanjutkan ke proses clustering yang mengadopsi proses clustering metode k-means. Posisi partikel akhir dari proses clustering menggunakan metode RCE direpresentasikan sebagai centroid pada K-means, sehingga posisi partikel pada tahap ini diperbarui seperti halnya penentuan centroid baru pada metode clustering K-means. Penetuan centroid pada K-means dilakukan menggunakan persamaan (8).

$$= - \sum_{\forall} \quad (8)$$

dimana x_p adalah titik data ke-p, z_j melambangkan titik pusat klaster ke-j dalam hal ini merepresentasikan partikel ke-j, dan d merupakan jumlah atribut dari tiap pusat klaster.

3. HASIL DAN PEMBAHASAN

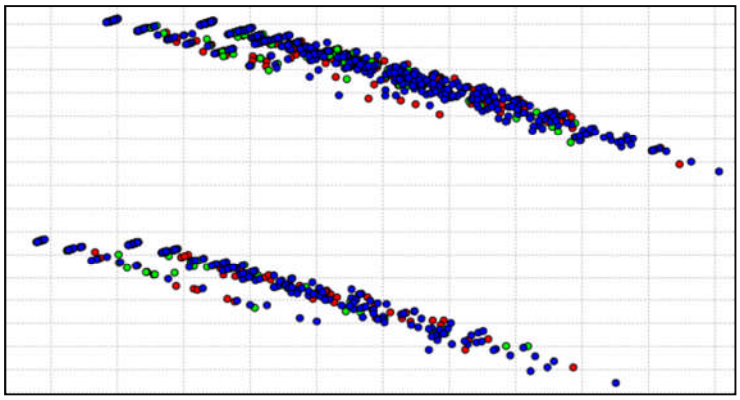
Metode clustering RCE-Kmeans diuji menggunakan 10 dataset. Skema hasil clustering yang diperoleh untuk masing-masing dataset kemudian dihitung kualitasnya menggunakan alat ukur *overall entropy*, *overall purity*, *dan percentage misclassification* [14]. Kualitas skema clustering dari metode RCE-Kmeans kemudian dibandingkan dengan kualitas skema clustering dari metode RCE dan K-means. Selain membandingkan nilai kualitas skema clustering, dibandingkan pula nilai standar deviasi dari ketiga metode tersebut. Rincian dataset yang digunakan pada penelitian ini ditunjukkan pada Tabel 2. Dataset diperoleh dari [16][17].

Tabel 3 menunjukkan nilai kualitas skema clustering yang dihasilkan metode K-means, RCE dan RCE-Kmeans. Pada dataset Iris, Wine, Glass, Dermatology, Yeast, Texture, dan Optical Digits metode RCE-Kmeans mampu menghasilkan nilai entropy yang lebih kecil dibandingkan dengan dua metode lainnya dan untuk dataset WDBC nilai entropy terendah dihasilkan oleh metode K-means. Hal ini berarti cluster-cluster yang dihasilkan oleh metode RCE-Kmeans lebih homogen dibandingkan 2 metode lainnya yaitu K-means dan RCE. Nilai *overall purity* dari klaster yang dihasilkan oleh metode clustering RCE-Kmeans untuk dataset Iris, Wine, Glass, Yeast, Texture, Optical Digits, dan Thyroid lebih besar dibandingkan dengan metode K-means dan RCE. Sedangkan jika nilai *overall purity* dari RCE-Kmeans dibandingkan dengan metode pendahulunya yaitu metode RCE maka metode RCE-Kmeans menghasilkan nilai *overall purity* klaster yang lebih besar dibandingkan dengan metode RCE untuk 9 data set kecuali untuk dataset CMC. Metode RCE-Kmeans menghasilkan nilai *percentage classification* yang terendah dibandingkan K-means dan RCE pada dataset Iris, Wine, Glass, Yeast, Texture, Optical Digits, dan Thyroid. Sedangkan nilai *percentage misclassification* terkecil untuk dataset CMC dihasilkan oleh metode RCE dan pada dataset WDBC dan Thyroid dihasilkan oleh metode K-means.

Tabel 2 Dataset

Dataset	Karakteristik		
	Jumlah Data	Atribut	Kelas
Iris	150	4	3
Wine	178	13	3
Glass	214	9	7
Dermatology	358	34	6
WDBC	569	30	2
CMC	1473	9	3
Yeast	1484	8	10
Texture	5500	40	11
Optical Digits	5620	64	10
Thyroid	7200	21	3

Standar deviasi kualitas skema clustering dari metode K-means, RCE, dan RCE-Kmeans ditunjukkan pada Tabel 4. Pada dataset Iris, Wine, Glass, Yeast, Texture dan Optical Digits metode RCE-Kmeans memiliki standar deviasi *overall entropy* terkecil. Pada dataset Dermatology dan WDBC metode clustering K-means memiliki standar deviasi entropy terkecil. Jika nilai standar deviasi *overall entropy* dari metode RCE dibandingkan dengan RCE-Kmeans maka terlihat bahwa metode RCE-Kmeans menghasilkan nilai yang lebih kecil untuk setiap dataset kecuali pada dataset CMC dan Thyroid dimana RCE dan RCE-Kmeans bernilai sama. Berdasarkan hasil perbandingan nilai *overall entropy* dan nilai standar deviasi *overall entropy* dapat diketahui bahwa cluster yang dihasilkan dari metode clustering RCE-Kmeans lebih homogen dan juga metode RCE-Kmeans dapat menghasilkan nilai standar yang lebih kecil dibandingkan K-means dan RCE. Berdasarkan standar deviasi *overall purity* metode RCE-Kmeans memiliki nilai terkecil untuk 5 dataset. Sedangkan apabila nilai standar deviasi *overall purity* RCE-Kmeans dibandingkan dengan RCE maka metode RCE-Kmeans selalu memberikan nilai yang lebih kecil untuk setiap dataset kecuali pada dataset Dermatology, CMC, dan Thyroid. Pada perbandingan standar deviasi *percentage missclassification* terlihat pada Tabel 3 bahwa metode RCE-Kmeans selalu menghasilkan standar deviasi yang terendah kecuali untuk dataset Dermatology, CMC, dan Thyroid standar deviasi *percentage misclassification* dari metode RCE-Kmeans selalu lebih kecil dibandingkan dengan metode RCE kecuali pada dataset Wine.



Gambar 2 Visualisasi sebaran data pada dataset CMC

Tabel 3 Kualitas skema clustering

Dataset	Overall Entropy (Eg)			Overall Purity (Pg)			Percentage Misclassification (Pm)		
	K-means	RCE	RCE-Kmeans	K-means	RCE	RCE-Kmeans	K-means	RCE	RCE-Kmeans
Iris	0.338	0.321	0.286	0.173	0.147	0.11	0.173	0.147	0.11
Wine	0.288	0.396	0.199	0.125	0.185	0.07	0.125	0.185	0.07
Glass	1.082	1.040	1.018	0.525	0.534	0.517	0.525	0.533	0.517
Dermatology	0.377	0.43	0.358	0.298	0.318	0.311	0.298	0.318	0.311
WDBC	0.258	0.3	0.261	0.072	0.104	0.074	0.072	0.104	0.074
CMC	1.036	1.037	1.036	0.615	0.607	0.609	0.615	0.607	0.609
Yeast	1.389	1.324	1.245	0.612	0.614	0.580	0.612	0.614	0.580
Texture	1.191	1.095	0.968	0.529	0.491	0.445	0.529	0.491	0.445
Optical Digits	0.804	0.904	0.699	0.333	0.387	0.285	0.333	0.387	0.285
Thyroid	0.306	0.305	0.305	0.359	0.361	0.320	0.359	0.361	0.320

Tabel 4 Nilai standar deviasi

Dataset	Standar deviasi overall entropy (Eg)			Standar deviasi overall purity (Pg)			Standar deviasi percentage misclassification (Pm)		
	K-means	RCE	RCE-Kmeans	K-means	RCE	RCE-Kmeans	K-means	RCE	RCE-Kmeans
Iris	0.101	0.077	0.008	0.124	0.079	0.006	0.124	0.079	0.006
Wine	0.191	0.154	0.109	0.132	0.121	0.082	0.132	0.121	0.082
Glass	0.082	0.071	0.051	0.035	0.04	0.032	0.035	0.04	0.033
Dermatology	0.161	0.189	0.175	0.120	0.118	0.126	0.120	0.118	0.126
WDBC	0.004	0.055	0.005	0.001	0.041	0.002	0.001	0.041	0.002
CMC	0.010	0.008	0.008	0.032	0.022	0.022	0.031	0.022	0.022
Yeast	0.085	0.083	0.076	0.036	0.054	0.037	0.037	0.054	0.037
Texture	0.161	0.187	0.079	0.059	0.082	0.048	0.06	0.082	0.048
Optical Digits	0.141	0.148	0.096	0.082	0.079	0.074	0.082	0.079	0.074
Thyroid	0.003	0.003	0.003	0.132	0.108	0.115	0.132	0.108	0.115

Beberapa dataset seperti CMC dan Yeast menghasilkan kualitas skema clustering yang lebih buruk dibandingkan dengan dataset lainnya. Hal ini terlihat dari tingginya nilai overall entropy, overall purity, dan percentage misclassification. Hal ini bisa saja disebabkan Yeast memiliki distribusi data yang *skewed*, yaitu dataset ini memiliki klaster dengan ukuran yang beragam. Berdasarkan clustering yang dilakukan pada dataset Yeast menggunakan metode K-means, RCE, dan RCE-Kmeans, diperoleh hasil yang menunjukkan klaster berukuran besar cenderung akan terbagi dalam beberapa klaster berbeda. Sedangkan pada dataset CMC walaupun ukuran klaster pada dataset CMC tidak bervariasi tetapi berdasarkan visualisai titik data yang ditunjukkan pada Gambar 2, dataset CMC memiliki bentuk klaster yang *elongated*, dimana bentuk klaster seperti ini cenderung dibagi menjadi *spherical cluster*. Hal ini yang menjadikan kualitas skema clustering untuk dataset CMC menjadi buruk. Adanya klaster-klaster yang *poor-separated* pada dataset juga mempengaruhi buruknya kualitas hasil clustering baik menggunakan metode K-means, RCE, dan RCE-Kmeans.

Durasi clustering untuk setiap metode ditunjukkan pada Tabel 5. Berdasarkan Tabel 5 tampak bahwa durasi clustering metode k-means bertambah berdasarkan jumlah data dalam dataset, jumlah atribut, dan jumlah kelas. Durasi clustering dari metode clustering RCE dan RCE-Kmeans tidak dipengaruhi secara signifikan oleh jumlah titik datanya. Hal ini disebabkan karena pada kedua metode tersebut terdapat faktor lain yang juga mempengaruhi durasi clustering yaitu adanya proses untuk perbaruan posisi partikel yang tidak pernah *win*. Sehingga

durasi dari proses clustering menggunakan metode RCE maupun RCE-Kmeans sangat bergantung pada ada tidaknya partikel yang tidak win dalam proses clusteringnya. Semakin banyak partikel yang tidak win durasi proses clustering semakin bertambah.

Tabel 5 Durasi proses clustering

Dataset	Metode		
	K-means	RCE	RCE-Kmeans
Iris	0.116	0.415	0.342
Wine	0.182	0.365	0.357
Glass	0.193	1.098	1.673
Dermatology	0.6	1.943	1.415
WDBC	0.4	1.183	0.913
CMC	0.4	1.016	1.045
Yeast	0.6	4.299	3.125
Texture	5.16	25.178	25.565
Optical Digits	20.42	27.068	22.775
Thyroid	1.87	4.414	3.868

4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan dan berdasarkan hasil pengujian metode RCE-Kmeans untuk masalah clustering data. Maka dapat ditarik kesimpulan sebagai berikut :

1. Penerapan metode K-means untuk memperbaiki pusat klaster yang dihasilkan metode RCE mampu menghasilkan klaster yang lebih padat. Hal ini terlihat dari dari nilai *average scattering* skema clustering metode RCE-Kmeans yang lebih kecil dibandingkan dengan RCE.
2. Kualitas skema clustering yang diukur menggunakan menggunakan overall entropy, overall purity dan percentage misclassification menunjukkan bahwa kualitas skema clustering yang dihasilkan oleh metode RCE-Kmeans lebih baik pada 7 dataset dibandingkan metode K-means dan dibandingkan dengan metode RCE lebih baik pada 8 dataset. Sedangkan berdasarkan durasi clustering metode K-means merupakan metode yang paling cepat dibandingkan RCE dan RCE-Kmeans.
3. Berdasarkan nilai standar deviasi kualitas skema clustering untuk setiap data set, metode RCE-Kmeans merupakan metode clustering yang mampu menghasilkan kualitas clustering yang lebih stabil, yaitu hasil clusteringnya tidak terpengaruh *initial state* seperti pada K-means atau *equilibrium state* seperti pada metode RCE.

5. SARAN

Saran untuk penelitian selanjutnya yang memanfaatkan metode RCE-Kmeans untuk memecahkan masalah clustering :

1. Pada penelitian ini parameter φ_1, φ_2 dan φ_3 bersifat statis, pada penelitian selanjutnya diharapkan adanya penerapan Time-varying acceleration co-effective (TVAC) untuk parameter φ_1, φ_2 dan φ_3.
2. Mencoba metode *local search* lainnya untuk melakukan penghalusan hasil clustering selain metode K-means.
3. Mencoba menggunakan fungsi jarak yang lain untuk menentukan *similarity* , seperti Mahalanobis atau menggunakan fungsi korelasi seperti Pearson dan Spearman untuk menentukan *similarity*.

DAFTAR PUSTAKA

[1] Žalik, K.R., 2008, An Efficient K'-means Clustering Algorithm, *Pattern Recognition Letters*, 29(9), pp.1385–1391.

[2] Yi, B., Yang, F., Qiao, H., Xu, C., 2010, An Improved Initialization Center Algorithm for K-means Clustering, *2010 International Conference on Computational Intelligence and Software Engineering (CiSE)*, (1), pp.1–4.

[3] Abul Hasan, M.J. dan Ramakrishnan, S., 2011, A Survey: Hybrid Evolutionary Algorithms For Cluster Analysis, *Artificial Intelligence Review*, 36(3), pp.179–204.

[4] Jain, A.K. dan Lansing, E., 2010, Data Clustering : 50 Years Beyond K-means, *Pattern Recognition Letters*, 31(8), pp.651-666.

[5] Kao, Y. dan Lee, S., 2009, Combining K-means and Particle Swarm Optimization for Dynamic Data Clustering Problems, *Computing and Intelligent Systems, ICIS*, (1), pp.757–761.

[6] Van der Merwe, D.W. dan Engelbrecht, a. P., 2003, Data clustering Using Particle Swarm Optimization, *The 2003 Congress on Evolutionary Computation (CEC)*, pp.215–220.

[7] Kao, Y.-T., Zahara, E., Kao, I.-W., 2008, A Hybridized Approach to Data Clustering. *Expert Systems with Applications*, 34(3), pp.1754–1762.

[8] Ye, F. dan Chen, C., 2005, Alternative KPSO-Clustering Algorithm, *Tamkang Journal of science and Engineering*, 8(2), pp.165–174.

[9] Eberhart, R. dan Kennedy, J., 1995, A New Optimizer Using Particle Swarm Theory, *IEEE the Sixth International Symposium on Micro Machine and Human Science*, pp.39–43.

[10] Shen, H., Jin, L., Zhu, Y., Zhu, Z., 2010, Hybridization of Particle Swarm Optimization with the K-means Algorithm for Clustering Analysis, *2010 IEEE Fifth International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA)*, pp.531–535.

[11] Ahmadyfard, A. dan Modares, H., 2008, Combining PSO and K-means to Enhance Data Clustering, *International Symposium on Telecommunications*, pp.688–691.

[12] Cohen, S.C.M. dan Castro, L.N. De, 2006, Data Clustering with Particle Swarms, *IEEE Congress on Evolutionary Computations*, pp.1792–1798.

[13] Szabo, A., Prior, A.K.F., Castro, L.N. De, 2010, The Proposal of a Velocity Memoryless Clustering Swarm, *Proc 2010 IEEE Congress on Evolutionary Computation (CEC)*, pp.1–5.

[14] Yuwono, M., Su, S.W., Moulton, B., Nguyen, H., 2012, Method for Increasing the Computation Speed of an Unsupervised Learning Approach for Data Clustering, *2012 IEEE Congress on Evolutionary Computation*, pp.1–8.

[15] Naik, B., Swetanisha, S., Behera, D.K., Mahapatra, S., Padhi, B.K., 2012, Cooperative Swarm Based Clustering Algorithm Based on PSO and K-means to Find Optimal Cluster Centroids. *IEEE National Conference on Computing and Communication System (NCCS)*, pp. 0-4.

[16] UCI Repository of Machine Learning Databases, On line Datsets, http://archive.ics.uci.edu/ml/ diakses 20 Maret 2014

[17] J. Alcalá-Fdez, A. Fernandez, J. Luengo, J. Derrac, S. García, L. Sánchez, F. Herrera. KEEL Data-Mining Software Tool: Data Set Repository, Integration of Algorithms and Experimental Analysis Framework. *Journal of Multiple-Valued Logic and Soft Computing*, 17:2-3 (2011) 255-287